1-15

PDP 1

COURSE WORKBOOK

DIGITAL EQUIPMENT CORPORATION · MAYNARD, MASSACHUSETTS

CBI 60 Box 3 39. E. 4. 4

Edmund C. Berkeley

INTRODUCTION

This workbook is designed to assist the student in understanding the programming of PDP-1 computer systems. When used in conjunction with the PDP-1 Handbook (F-15D), it serves as a guide through the course and provides a reference for all the material covered.

The contents of the workbook include text material describing the PDP-1, illustrative problems in programming, and homework assignments. Since each area covered in the course is dependent on knowledge of the preceding topics, it is strongly suggested that the student complete the problems at the end of a chapter to insure full understanding of each subject.

The course consists of a series of lectures and laboratory periods to aid the student in understanding the operation of the PDP-1 system. During these periods, the uses and capabilities of "software" packages of interest to the programmer are discussed. By the end of the course each student should be able to employ all the equipment in the PDP-1 system.

CBI 60 Bx 3 39.E.4 A

TABLE OF CONTENTS

Page

INTRODUCTION

CHAPTER I

The Concept of a Digital Computer 1-1

CHAPTER II

The Concepts of Computer Programming 2-1

CHAPTER III

Introduction to the PDP-1 Systems 3-1

CHAPTER IV

Basic PDP-1 Instruction List 4-1

CHAPTER V

Introduction to MACRO Assembly Program 5-1

CHAPTER VI

Introduction to the PDP-1 Symbolic Tape Editor........... 6-1

CHAPTER VII

Introduction to the PDP-1 Debugging Tape............... 7-1

CHAPTER VIII

Input-Output Transfers 8-1

CHAPTER IX

Single Channel Sequence Break System 9-1

The first example creates a new dummy symbol, and the second reassigns an existing one for subsequent appearances.

C. FORMAT OF THE SOURCE LANGUAGE TAPE

The source language tape can be produced off-line with a Friden Flexowriter using the FIO-DEC code (see Handbook, page 68) or with the use of an on-line program such as "expensive Typewriter".

The symbolic source language program, as typed on paper, should appear in the following form. The first line of all symbolic programs must be a title line. The title line is terminated with a carriage return. A good format for the body of the program is the following:

```
address tag, ————————→| instruction ————————→|/comment
DO NOTHING
beg, ————————→| lac (34   ————————→| /use of a constant
     ————————→| add tem̄   ————————→| /use of a variable
     ————————→| hlt
constants
variables
start beg ↓
(stop code)
```

In the previous example, the title appears as the first line. It is followed by the body of the source program. The body is divided into three columns:

- address tags (if used)
- instruction mnemonics with numbers or symbolic addresses
- comments (if used)

The three columns are separated by tabs. If this format is adhered to throughout the body of the program, it is very easy to read the listing of the program. Since a constant and variable are used in the program, the pseudo instructions "constants" and "variables" appear following the body of the program (it is the option of the programmer that the pseudo instructions appear here even though this is the usual case). The last line of the symbolic program is a start line and an address. This gives the address where the program is to start running.

Other formats may be used; however, this format is the one followed in this workbook.

EXAMPLE: Write a routine to clear all of memory (excluding the portion containing the program) in the standard PDP-1.

SOLUTION: Although there are always a variety of ways to write any program, the most efficient way in this case is given below.

```
ZERO MEMORY
0/
beg,      dzm .+3.                0/ 340003
          idx beg                 1/ 440000
          jmp beg                 2/ 600000
start beg ↲
(stop code)
```

In the above example, the symbolic version is shown at the left. The binary output tape appears as that shown at the right and reads into the memory registers indicated. The jump block has "jmp 0000" punched in it. After reading in, the program automatically begins by executing the instruction in location 0000.

The same instruction steps act somewhat differently when executed at other locations in memory as in the following example:

```
ZERO MEMORY
7775/
beg,                dzm 0
                    idx beg
                    jmp beg
start beg
(stop code)
```

It is recommended that the student analyze the two above examples. Do the programs clear themselves? Do the routines destroy themselves? Do the programs halt to continue running indefinitely?

D. OPERATING INSTRUCTIONS FOR MACRO

Macro uses the PDP-1 tape reader, punch, and typewriter; therefore, these mechanisms should be properly initialized before starting an assembly. All tapes should be placed in the reader with the feed hole toward the computer, and the reader must always be turned on before the

CHAPTER VI

INTRODUCTION TO THE PDP-1 SYMBOLIC TAPE EDITOR

("Expensive Typewriter" Digital 1-4-S)

The PDP-1 symbolic tape editor allows the programmer to produce or correct symbolic tapes with the PDP-1 computer if an off-line device such as a Flexowriter is not available.

"Expensive Typewriter" starts in location 100_8 and is controlled by the on-line typewriter. There are two types of inputs available to the programmer. "Expensive Typewriter" can read paper tape into its buffer and accept typewriter characters. Upon command, the contents of its buffer can be punched as a FIO-DEC tape.

There are two basic modes of operation. In the control mode, characters typed are commands to "Expensive Typewriter" to perform some function: in the text mode, all typed input (except backspace) is copied directly into the buffer. Control mode characters print in red, and text mode characters print in black.

There is room in the buffer of "Expensive Typewriter" for approximately 8,000 characters. This buffer is intended to hold a page of typescript, organized as a list of lines. Each line can be called by a decimal number counting from the top of a page. All lines fed into the buffer must end with a carriage return.

The following list contains the important control characters which are used by the operator when in the control mode.

a	Enter the text mode and append the following typed text onto the end of the buffer.
Ni	insert following text before line N.
Nc	Change line N from old text to that following.
Nd	Delete line N. Successive lines are moved up in buffer to fill gap.
Nl	List or print out line N.
p	Punch the contents of the stored text as a FIO-DEC tape. The contents of the buffer remain unchanged.
s	Punch a stop code (should be used after punching each page).
k	Kill or clear entire buffer.

r	Read a symbolic tape (reader must be on) and enter it into the buffer. Reading stops when a stop code is encountered.
Nt	Duplicate N pages of tape. Same as repeating the following sequence of commands N times: k, r, p, s, k.
w	Write or list contents of entire buffer.
carr. ret.	Execute last typed-in command.
tab	Execute last typed-in command.
. (middle dot)	Cancel last typed-in command (must be given before tab or carriage return).
/ (slash)	Equal to number of last line in buffer.
. (period)	Equal to number of last line referred to.
space	Add
- (minus)	Subtract
, (comma)	Separates two decimal numbers (can be used to change, delete or list a group of lines, eg., N,Md).
backspace	List or print out the line following that just referred to (same as . 1l).

In the text mode all characters are stored in the buffer; however, the following two have a special meaning.

backspace	Delete last character typed as text.
carr. ret.	Terminate a line of text, store it in buffer, and advance line counter.

When in the text mode, the first three SENSE SWITCHes have the following meaning:

SS1 on, then off	Indicates end of text mode and returns program to control mode.
SS2 on	Causes the line numbers to be printed when in "w" mode.
SS3 on	Causes backspaces to be entered as normal characters.

When in the control mode, SENSE SWITCH 1 has the following meaning:

SS1 on	When reading tape, suppress parity checking and enter blank tape as spaces. When printing or punching, stop output immediately, print a carriage return, and enter control mode.